

# **NTS: Programming Languages and Paradigms**

**Jiří Zlatuška**

A new, more powerful successor  
to T<sub>E</sub>X-the-typesetter.

Objectives



```
graph LR; A[A new, more powerful successor to T_E X-the-typesetter.] <-- B[Objectives]
```

Objectives

```
graph LR; A[Objectives] --> B["A new, more powerful successor to TeX-the-typesetter."]; A --> C["An effort to re-program TeX-the-program."];
```

A new, more powerful successor  
to TeX-the-typesetter.

An effort to re-program TeX-the-program.

A procedural language based on a formal syntax and well-defined semantics.

Objectives

Pascal



A procedural language based on a formal syntax and well-defined semantics.

Objectives

Structured programming as a methodology.

Pascal

A procedural language based on a formal syntax and well-defined semantics.

Objectives

Structured programming as a methodology.

Pascal

Building large programs from manageable smaller components.

A procedural language based on a formal syntax and well-defined semantics.

Objectives

Structured programming as a methodology.

Pascal

Building large programs from manageable smaller components.

Abstract data structures instead of just the data types provided by architectures.

A procedural language based on a formal syntax and well-defined semantics.

Objectives

Structured programming as a methodology.

Pascal

Building large programs from manageable smaller components.

Abstract data structures instead of just the data types provided by architectures.

Portability to different platforms.



Introducing the world to literate programming.

Objectives

Pascal

Knuth's  
choices

Introducing the world to literate programming.

Extending Pascal by pre-processing the source code.

Objectives

Pascal

Knuth's choices

Introducing the world to literate programming.

Extending Pascal by pre-processing the source code.

Tangling a WEB and weaving the documentation.

Objectives

Pascal

Knuth's choices

Introducing the world to literate programming.

Extending Pascal by pre-processing the source code.

Tangling a WEB and weaving the documentation.

Arithmetic calculations and memory management built in to guarantee portability.

Objectives

Pascal

Knuth's choices

Extending  $\text{T}_{\text{E}}\text{X}$  is not that easy.

Objectives

Pascal

Knuth's  
choices

20 years later

Extending  $\text{T}_{\text{E}}\text{X}$  is not that easy.

Successors of Pascal didn't make it.

Objectives

Pascal

Knuth's  
choices

20 years later

Extending  $\text{T}_{\text{E}}\text{X}$  is not that easy.

Successors of Pascal didn't make it.

Even Knuth has moved to C.

Objectives

Pascal

Knuth's  
choices

20 years later

Extending  $\text{T}_{\text{E}}\text{X}$  is not that easy.

Successors of Pascal didn't make it.

Even Knuth has moved to C.

Hardware is no longer a constraint.

Objectives

Pascal

Knuth's  
choices

20 years later



Extending  $\text{\TeX}$  is not that easy.

Successors of Pascal didn't make it.

Even Knuth has moved to C.

Hardware is no longer a constraint.

$\text{\TeX}$  data structures have become time bombs.

Objectives

Pascal

Knuth's  
choices

20 years later

Extending T<sub>E</sub>X is not that easy.

Successors of Pascal didn't make it.

Even Knuth has moved to C.

Hardware is no longer a constraint.

T<sub>E</sub>X data structures have become time bombs.

T<sub>E</sub>X's monolithic, interwoven character has become a burden.

Objectives

Pascal

Knuth's choices

20 years later

Extending  $\text{T}_{\text{E}}\text{X}$  is not that easy.

Successors of Pascal didn't make it.

Even Knuth has moved to C.

Hardware is no longer a constraint.

$\text{T}_{\text{E}}\text{X}$  data structures have become time bombs.

$\text{T}_{\text{E}}\text{X}$ 's monolithic, interwoven character has become a burden.

We need to re-create  $\text{T}_{\text{E}}\text{X}$  using modern methodologies and technologies.

Objectives

Pascal

Knuth's choices

20 years later

Removing all constraints and most complexity.

Objectives

Pascal

Knuth's  
choices

20 years later

Options



Removing all constraints and most complexity.

Adding modularity and clear interfaces.

Objectives

Pascal

Knuth's  
choices

20 years later

Options

Removing all constraints and most complexity.

Adding modularity and clear interfaces.

Functional programming would suit  
T<sub>E</sub>X's basic manipulations: Lisp?

Objectives

Pascal

Knuth's  
choices

20 years later

Options

Removing all constraints and most complexity.

Adding modularity and clear interfaces.

Functional programming would suit  
T<sub>E</sub>X's basic manipulations: Lisp?

Logic programming would help us specify  
and solve typographic problems: Prolog?

Objectives

Pascal

Knuth's  
choices

20 years later

Options

Removing all constraints and most complexity.

Adding modularity and clear interfaces.

Functional programming would suit  
 $\text{T}_{\text{E}}\text{X}$ 's basic manipulations: Lisp?

Logic programming would help us specify  
and solve typographic problems: Prolog?

Procedural programming along the  
lines  $\text{T}_{\text{E}}\text{X}$  is coded now: C and C++?

Objectives

Pascal

Knuth's  
choices

20 years later

Options



Removing all constraints and most complexity.

Adding modularity and clear interfaces.

Functional programming would suit  
 $\text{T}_{\text{E}}\text{X}$ 's basic manipulations: Lisp?

Logic programming would help us specify  
and solve typographic problems: Prolog?

Procedural programming along the  
lines  $\text{T}_{\text{E}}\text{X}$  is coded now: C and C++?

The portable compromise, procedural  
programming while using objects: Java!

Objectives

Pascal

Knuth's  
choices

20 years later

Options

Provide a high level of structure  
using objects and methods.

Objectives

Pascal

Knuth's  
choices

20 years later

Options

NTS in Java

Provide a high level of structure  
using objects and methods.

Anchor NTS in the World Wide Web.

Objectives

Pascal

Knuth's  
choices

20 years later

Options

NTS in Java

Provide a high level of structure  
using objects and methods.

Anchor NTS in the World Wide Web.

Support as many platforms as possible.

Objectives

Pascal

Knuth's  
choices

20 years later

Options

NTS in Java

Provide a high level of structure  
using objects and methods.

Anchor NTS in the World Wide Web.

Support as many platforms as possible.

Use standardized interfaces.

Objectives

Pascal

Knuth's  
choices

20 years later

Options

NTS in Java

Provide a high level of structure  
using objects and methods.

Anchor NTS in the World Wide Web.

Support as many platforms as possible.

Use standardized interfaces.

Move from WEB to WWWeb.

Objectives

Pascal

Knuth's  
choices

20 years later

Options

NTS in Java